4SL4 Assignment 3

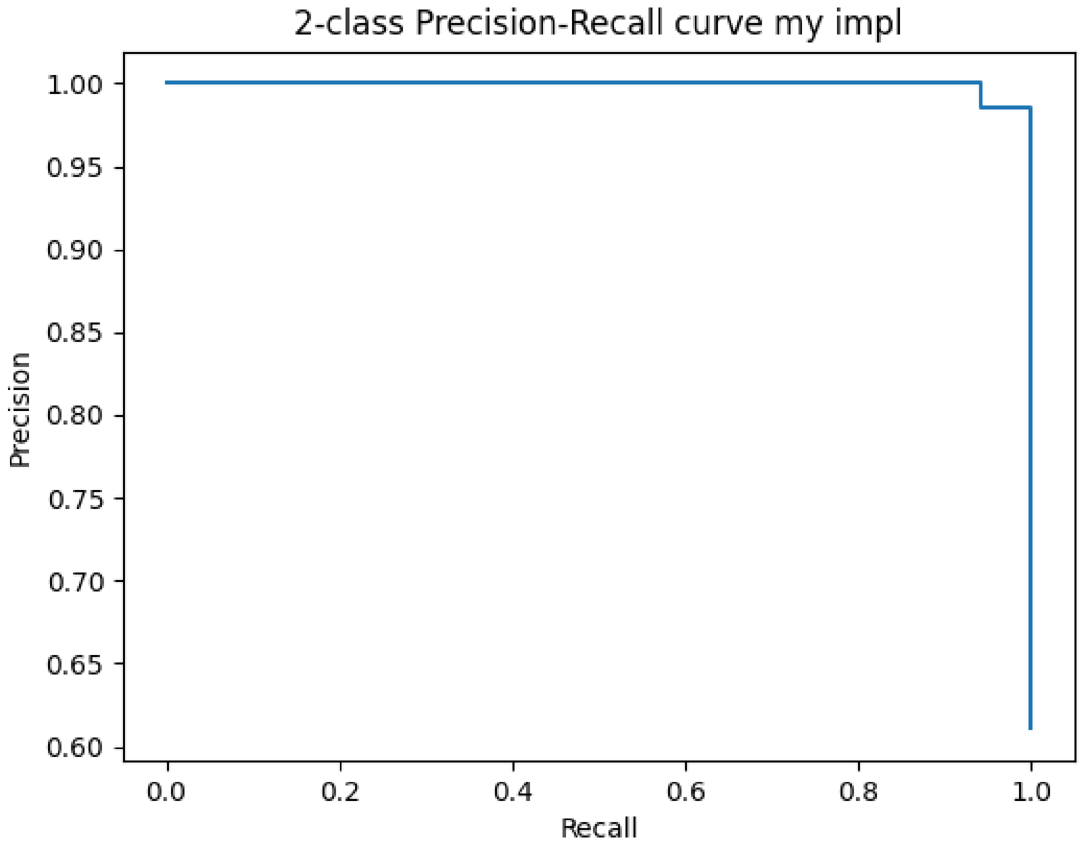
Aaron Pinto
400190637
pintoa9

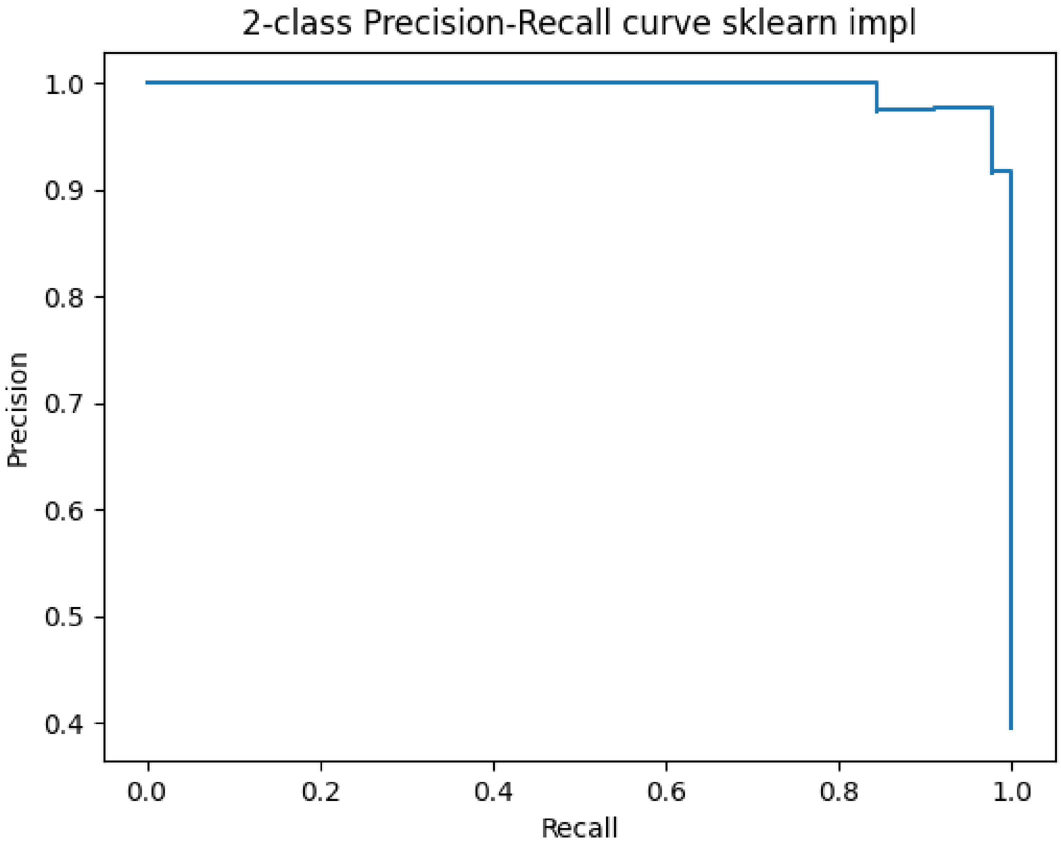
Model	Test Error (Misclassification rate)	F1-score
My logistic regression	0.008771929824561403	0.9928057553956835
Sklearn logistic regression	0.017543859649122806	0.9855072463768116
My k-nearest neighbours	0.043859649122807015	0.9640287769784173
Sklearn k-nearest neighbours	0.043859649122807015	0.9640287769784173

My results are the exact same as scikit-learn for the k-nearest neighbours classifier. They are very slightly different compared to scikit-learn for the logistic regression classifier, most likely because of different algorithms used to solve the regression model. The scikit learn implementation also uses L2 regularization for the weights by default. The best model overall is my implementation of the logistic regression, in both the misclassification rate and F1 score metrics.

Logistic Regression

Logistic regression model	Parameter Vector
Mine	[0.38578814 -0.37943891 -0.38390586 -0.37426186 -0.38885302 -0.14267871 -0.09386633 -0.35941573 -0.40689446 -0.08581256 0.16239294 -0.38466637 -0.01618699 -0.32772219 -0.36056542 -0.01106973 0.15258689 -0.02940999 -0.07581566 0.08330069 0.19700897 -0.46663668 -0.45958923 -0.44610123 -0.45575449 -0.34213066 -0.1725378 -0.36048119 -0.42516782 -0.27508916 -0.13189663]
Scikit learn	[0.37463202 -0.35308871 -0.54730243 -0.32359993 -0.41442887 -0.17301272 0.69450415 -0.68815488 -0.83824172 0.08441986 0.1722472 -1.1664609 0.11898396 -0.5408113 -0.96719992 -0.1873305 0.79723467 -0.3754317 -0.25516416 0.18540209 0.77841921 -0.92955578 -1.00379385 -0.6947039 -0.93456161 -0.80756397 0.15823116 -0.84925601 -0.81914754 -0.63164475 -0.73204015]


2-class Precision-Recall curve my impl
Precision
1.00
0.95
0.90
0.85
0.80
0.75
0.70
0.65
0.60
0.0
0.2
0.4
0.6
0.8
1.0
Recall


2-class Precision-Recall curve sklearn impl
Precision
1.0
0.9
0.8
0.7
0.6
0.5
0.4
Recall
0.0
0.2
0.4
0.6
0.8
1.0

k-Nearest Neighbours

I used 5-fold cross-validation, because that’s what suggested here https://scikit-learn.org/stable/modules/cross_validation.html.

To deal with the situations in which ties between distances occur, I chose to pick the neighbour depending on its position in the training dataset. Whichever one had a lower row index is the one I picked. This is random because train_test_split() shuffles the data at the start anyway.

To deal with the situations with even k, where the number of classes of each type is split 50/50, I erred on the side of caution and chose to go with the malignant class, because it is safer to report a malignant tumour when it is actually benign vs reporting a benign tumour when it is actually malignant, because malignant tumours are much worse than benign tumours.

K	Cross-validation error
1	0.032967032967032975
2	0.05274725274725275
3	0.03076923076923077
4	0.032967032967032975
5	0.032967032967032975